Utilizing Free Software to Design and Maintain a Secure Digital Network

Logan J. Erbst

Edited by: David McMackins

May, 2016

Abstract

The presence of proprietary software on computer systems today is the cause of many of the problems computer users are concerned about. For instance, with many proprietary programs being turned into subscription-based services, users face losing access to their software and data. Others worry about security vulnerabilities and government spying. The developers of proprietary software create malicious features with which they breed user dependence on their software in the name of profits. By replacing that software with free software, users can enjoy the benefits of computer technology without sacrificing freedom, security, and privacy. By understanding that freedom should be valued over convenience, computer users can create a better world for themselves. This paper explains methods that can be used to secure one's computer systems and networks using free software and the reasons why continuing to use the proprietary alternatives are detrimental to the users.

Table of Contents

Utilizing Free Software to Design and Maintain a Secure Digital Network

Anybody that uses an electronic device is using some kind of software. That software can either be free or proprietary. The majority of software is proprietary, and this kind of software forces the user into an unjust agreement. These agreements are essentially legal handcuffs. Many proprietary end user license agreements also include Digital Rights/Restrictions Management (DRM), and DRM is a form of digital handcuffs. What kind of person wants to be handcuffed? **“**Proprietary software, also called nonfree software, means software that doesn't respect users' freedom and community. This means that its developer or owner has power over its users. This power is itself an injustice” (Gnu.org, 2015).

> “‘Free software’ means software that respects users' freedom and community. Roughly, it means that **the users have the freedom to run, copy, distribute, study, change and improve the software**. Thus, ‘free software’ is a matter of liberty, not price. To understand the concept, you should think of ‘free’ as in ‘free speech,’ not as in ‘free beer’. We sometimes call it ‘libre software’ to show we do not mean it is gratis” (FSF, 2016).

Power is a corrupting force, and this tempts developers to insert features into their programs which mistreat the users of any kind of computing device, and this makes it malware. (The term malware essentially means: software whose function is to mistreat the user or to steal information from the user.) “Of course, the developer usually does not do this out of malice, but rather to put the users at a disadvantage. That does not make it any less nasty or more legitimate,” (Gnu.org, 2015).

Because of this, you would generally want to use free software on your computers wherever possible. There may also be complications with licensing, as many companies have very strict licensing terms. These licenses can mean that the person and/or company that purchases the software doesn’t

actually own the software, but is rather ‘allowed’ to use it until the company that created the software decides to revoke said permissions.

Not to mention Microsoft is (as of the time of this writing) forcing users of their Windows 7 and 8 operating systems to download the Windows 10 installer. This is possible to prevent in a corporate setting with Windows Server Update Services (WSUS), but for the majority of small businesses and home users, they will have a 6 Gigabytes (GB) of data downloaded to their computer without their consent. Users with data limits imposed by their ISPs or with slow connections will be drastically affected because they will either have a larger bill to pay due to overages or they will be unable to use the Internet due to the unauthorized download.

> “Microsoft has long promised Windows 7 and 8.1 customers they would have a chance to opt out of the Win10 upgrade installation. It appears that accepting the EULA for Windows 10 is the only checkpoint -- if you accept the EULA, Windows 10 gets installed on your PC. Since most Windows users are accustomed to accepting EULAs (when's the last time you declined a EULA?), we're in for a flood of new Windows 10 users as systems reboot over the next few days.
>
> Not accepting the EULA doesn't disable the upgrade, though. As best I can tell (these are still early days), the EULA comes back on every reboot, even if you turned it down previously. Once the "Upgrade to Windows 10" update appears as checked on your PC, you're stuck in an endless installer cycle that kicks in every time you reboot. All of the installation files stay on your system, and Windows 10 is ready to be installed, again and again” (Leonhard, 2016).

Things like this are why people and companies need to remove Microsoft from their life. You can do nearly everything on a GNU/Linux system that you can do on a Windows system. This includes,

but is not limited to, productivity, software development, gaming, browsing the Internet, and reading your emails. Considering that the majority of people spend all their time in a web browser anymore, this furthers the argument that you don't need Windows on your computer systems anymore.

Now onto the hardware side of things: the best bet to liberate the routers, switches, and firewall appliances is to reuse older server boxes for the task. A company could use an older server box, with minimal specs (like an older single core Xeon) as a router or firewall appliance, and for a switch you could possibly use an older server box as well, but they would have to install multiple network cards into it, and then there would be possible bottlenecking on the network with the server acting as a switch. As of the time of this writing, the closest that a user/company can get to a liberated switch would be through opencompute.org or routerboard.com.

Of course the way that a company is going to want to lay out their network will vary based on the number of hardware devices, number of users, and desired bandwidth. There is one thing that should always remain the same, and that is the order of network placement of the firewall, router, and switches. Every company should have those devices in this order on every network segment (that is: a location with its own Internet connection).

Let us focus now on the most critical aspect of any network: the servers. With Microsoft products, you are lucky to get a patch for a critical vulnerability within a month of discovery, while with a free software system, patches are usually released within the day that they are discovered. Remember the Badlock bug? (CVE-2016-2118 {Samba in general} CVE-2016-0128 / MS16-047 {Windows}) GNU/Linux distributions like Debian had a fix for that the next day, while it took Microsoft a week or two to release their patch.

I've yet to mention the amount of malware for Windows systems. It has come to the point that if you have a Windows computer attached to the Internet, it is almost guaranteed to have some kind of

malware, and if it runs any version of Windows, the operating system itself is malware. The biggest threat that users and server operators experience when using Windows is the fact that they work closely with the NSA. This is because they tend to spy on users and collect data on them whether they like it or not. They report bugs and other user information to the NSA before they patch it on their systems, or the release a “patch” that supposedly fixes a security hole only to have the NSA and other government agencies exploiting it on a regular basis.

> “Companies and governments buy Microsoft's software, depending on the company to create programs that are secure and safe. No software is completely bug-free, and serious flaws are frequently found in Microsoft's code (and in open source, too, of course.) So the issue is not about whether software has flaws – every non-trivial piece of code does – but how the people who produce that code respond to them.
>
> What companies and governments want is for those flaws to be fixed as soon as possible, so that they can't be exploited by criminals to wreak damage on their systems. And yet we now learn that one of the first things that Microsoft does is to send information about those vulnerabilities to ‘multiple agencies’ - presumably that includes the NSA and CIA. Moreover, we also know that ‘this type of early alert allowed the U.S. to exploit vulnerabilities in software sold to foreign governments’”. (Moody, 2013)

Would any level-headed Information Technology specialist really want to put their trust in a company that reveals vulnerabilities to a government agency that has been known to violate the trust of the American people by spying on them? This is why more computers (servers as well) need to be running free software, not just to avoid government spying, but for the freedom of the users as well.

Companies like Microsoft and Apple may say that they care about their customers, but in reality, they truly only care about their profits. These corporations have marketing professionals on their payroll whose job is to claim things along the lines of ‘our customers are important to us, and that’s why we are <input marketing buzzwords here>’ in order to make the public view the company positively. These marketing teams know how to word things to make the most sales, even if the product is completely unusable.

Microsoft’s software has numerous backdoors as well. Let’s say a company buys a collection of new Windows 10 computers, and they either cannot afford the Pro version or forget to pay for the ‘enhanced’ versions. They enable “Device Encryption” on those computers, but they still are not truly safe, because Microsoft has their encryption key. This is in no way safe, as any sort of external possession of encryption keys is a security hole, and Microsoft Corporation has not had the best track record for server security.

“The fact that new Windows devices require users to backup their recovery key on Microsoft’s servers is remarkably similar to a key escrow system, but with an important difference” (Lee, 2015). After the automatic upload of the key to Microsoft’s servers the user can decide to ‘delete’ it from their servers. But once the user clicks the delete button, they can never really be sure that the file has indeed been deleted. For all the user knows, the file is still on the server, just unavailable to them.

> “Windows 10 comes with 13 screens of snooping options, all enabled by default, and turning them off would be daunting to most users.
>
> Windows 10 ships with default settings that show no regard for the privacy of its users, giving Microsoft the “right” to snoop on the users' files, text input, voice input, location info, contacts,

> calendar records and web browsing history, as well as automatically connecting the machines to open hotspots and showing targeted ads.
>
> Windows 10 sends identifiable information to Microsoft, even if a user turns off its Bing search and Cortana features, and activates the privacy-protection settings."(FSF Webmasters, 2016).

One way to know for certain that files that are uploaded to a remote server are inaccessible to the server operator is to encrypt it using an open standard like GPG (GNU Privacy Guard), which comes preinstalled on many free software operating systems. The users of non-free operating systems are often mistreated by the corporations that own them, all in the name of profits. Some of these corporations may say that they do some of the things that they do in the name of security, but what can the user do when the company that controls their computer begins to sell their information to advertising companies?

> "Microsoft admits to collecting information to personalize your experience, but says it does not scan your email to collect that. "Unlike some other platforms, no matter what privacy options you choose, neither Windows 10 nor any other Microsoft software scans the content of your email or other communications, or your files, in order to deliver targeted advertising to you," Microsoft senior vice president Terry Myerson wrote in a blog post" (Hachman, 2015).

Since 2007, Microsoft has been forcing unwanted changes on users' computers. Even when the user has disabled automatic updates, the system continues to run updates automatically. This system behavior is very much like the way many spyware programs work. Ironically, this kind of software is what "Windows Defender" is supposed to protect users from, yet Microsoft is creating malware.

There is really only one way to avoid the surveillance, the forced software installations, and any other problems caused by non-free software is to avoid it completely. Free systems like those approved

by the Free Software Foundation (ex: Trisquel GNU/Linux) do not force anything upon the users or spy on users of the system. If any such behavior is discovered, it is reported as a bug and removed from the system.

A company does not need Microsoft's centralized network authentication system known as Active Directory. Free software systems can use an open variant known as openLDAP combined with Kerberos. Active Directory has many additional settings for administering Windows Systems, but if a company is operating on a fully free software system, they really don't need those functionalities.

There is one major drawback of using openLDAP and that is the fact that it doesn't have a graphical user interface; the entire configuration is done via the terminal (Command Line). There are graphical utilities available, but it is recommended to either write a tool or learn the terminal. This could of course be changed if one were willing to pay for the development of a GUI application.

For a network firewall, there are many non-free systems that will perform this task, but anyone with any free software sense they will see that as an abomination.

> "IPFire is based on Linux, which is the best Open Source kernel around. Additionally, IPFire is **not** based on any other distribution like Knoppix is on Debian. It is compiled from the sources of every single package. This consumes a lot of work, but finally gives the opportunity to not rely on the update cycles of others. The advantages we gain is that we are able to select very stable versions of software and build the distribution from them. For example is the most part of the distribution quite well tested and long maintained - in contrast to the kernel which is very recent and regularly updated with patches to support as much hardware as possible and more importantly fix security errors" (IPfire, 2016).

IPfire is not the only thing needed for a secure free software network, but it is the beginning of one. It is a first step that any household or company can take. There is a lot that can be done to secure any network; the administrators and users just need to be educated.

Design

A properly secured free software network, for one thing, contains no non-free software (wherever possible). This can mean sticking with older hardware to avoid the non-free firmware required by newer components. Intel has had a good track record when it comes to ‘open source,’ which is a good thing, but their newer chips have a hardware backdoor. This backdoor is a processor chip hidden in their Central Processing Units (CPUs) that were/are made from 2011 onwards.

> “AMT is an auxiliary processor built into the high-end Intel Q chipsets with an i5 or i7 CPU. We don't know whether it is present in the cheaper H, Z, and B chipsets. It runs software loaded from a binary blob at an early stage in the process of booting the machine” (Vandewege, Garrett, & Stallman, 2014).

AMD has something similar to Intel ME, they call it the AMD Platform Security Processor (PSP). “The Platform Security Processor (PSP) is built in on all Family 16h + systems (basically anything post-2013) and controls the main x86 core startup. PSP firmware is cryptographically signed with a strong key similar to the Intel ME. If the PSP firmware is not present, or if the AMD signing key is not present, the x86 cores will not be released from reset, rendering the system inoperable” (Libreboot).

This means that you are stuck with old hardware right? Not exactly. The POWER CPUs from IBM are freedom friendly. IBM is openly attacking Intel for their proprietary way of thinking. These systems are open from the lowest level. The downside to them is that they are very expensive. “The new Power8 servers are available from October 31. Pricing starts at around $8,000 for a single-socket S812L; for a 16-socket system, you’re looking at hundreds of thousands of dollars, I imagine. (If a single Power8 chip was available for OEMs, it would probably be priced at around $5,000)” (Anthony, 2014).

Not only does the IBM POWER8 architecture respect users’ freedom, but has a much better performance per watt ratio than any Intel chip. In a truly freedom respecting system, everything proprietary is thrown out and is replaced with something free and open. That is on the server side of things. What about the workstation? Raptor Engineering has got that covered with their Talos Workstations.

> “Talos™ is the world's first ATX-compatible, workstation-class mainboard for the new, free-software friendly IBM POWER8 processor and architecture. Designed for security-conscious users requiring high performance, the flexible and extensible Talos™ mainboard includes two Coherent Accelerator Processor Interface (CAPI) capable slots; utilizes libre-toolchain FPGAs for system control and routing; provides a plethora of PCI Express slots; and includes a GPIO header for custom peripherals. Talos™ schematics and libre (fully open and auditable) firmware also are included” (Raptor Engineering, 2016).

Remember, freedom isn’t free; people must be willing to give up conveniences for freedom. The cheapest option (hardware wise), while the most convenient, is often not the most free while at the same time, the most expensive option is often not the most free.

Implementation

We know that the beginning of a secure network starts with IPfire, but what about the rest of the network? The answer to this is GNU/Linux, not just on the servers, but on the workstations as well. In reality, if there are no Windows or OS X systems on the network, it is safer than most. In many scenarios, having a 100% secure network is impossible, but 80% is reachable with proper precautions. First off, on the firewall side of things (through IPfire preferably), block off all unused ports; for instance, if all that is done on the network is web browsing, the only ports that would have allowed traffic is ports 80 and 443. The port whitelist will be different based upon each use-case.

Next thing is on the workstations: have a standardized set of applications that each system will use. This will be chosen based (once again) on what the network segment will need. Common applications are actually preinstalled on most GNU/Linux distributions, but each network segment will have applications that only they will need.

To increase operational security, a company should always locally host their own email servers, and those servers should be encrypted, generally using LUKS on GNU/Linux systems. This security can have an inconvenience, and that is that every time that the server needs to be rebooted, someone has to physically be at the system to type in the password (LUKS passwords should be no shorter than 32 characters for proper password security).

In the corporate world, two GNU/Linux distributions stand out, and those are Red Hat and SUSE. These systems include non-free software in the kernel itself, but Debian is a free-software system, when the “contrib” and “non-free” repositories are not used. If a package does not exist in Debian’s “main” repository, it is not free software according to the Debian Free Software Guidelines (DFSG). Therefore, Debian is a good system to use for servers and workstations. If someone truly is worried about the possibility of non-free software sneaking on their systems, then an FSF-endorsed system should be used. Trisquel GNU/Linux is one such system.

> “Trisquel GNU/Linux is a distribution of the GNU operating system, with the Linux-libre kernel. It comes ready for home and office use, and new programs are easy to find and install. Our Documentation will help you explore your options” (Trisquel, 2009).

These free-software systems that are approved by the FSF will not try to or ask you to install or use non-free software. They are also not guaranteed to work on the newest hardware. Users and sysadmins are going to want to stick with older hardware so that the free drivers will have a better chance at working properly with the hardware. This is the main limitation of free software: the lack of support for the newest hardware.

There is quite a lot of wireless networking hardware that refuses to function without non-free blobs. A blob in Linux in this case is a piece of non-free software, usually firmware that is loaded after the system is booted. These pieces of firmware used to be hard-coded into the hardware; why did the manufacturers ever decide that loading the firmware after the fact was ever a good idea?

Be warned, though, that free-software is not always simple to configure. While many non-free systems have simple 'point and click' interfaces, free-software usually requires the manual editing of a config file, which is usually a text file of some sort. Sysadmins can often spend several hours poring over a man (manual) file to figure out how to properly edit a config file, only to have to go back and edit it again because they either forgot something or they mistyped something.

With the additional security and software freedoms come numerous compromises. What is given up is usually ease of use and games, but in a business environment, games are useless, and ease of use is improved as more people put resources toward improvement of these free systems and programs. Remember that freedom has never been free, and it never will be. People must learn this, or they will lose their freedoms, and will have to fight tooth and nail to get them back.

Support

Supporting a free-software system is often quite difficult and can often lead sysadmins to return to the non-free options due to their simplicity in administration. Microsoft and Apple work to make system administration easy, but this simplicity comes at the cost of security. How secure is a non-free system? Only the developers really know.

Support on GNU/Linux usually involves editing some kind of config file on a server. On the client side, there really isn't too much. If the server is providing network authentication, that adds additional complexity, but it reduces needed support on the clients.

Client support is minimal, unless the user ends up breaking something in their user directory, then that can be a hassle to fix, depending on the software that is installed on the system. That is where standardized package sets can come in handy. To make it even easier the server, or a set of servers, can host a local software repository that is synced with the main repositories, this is usually done with rsync via a cron job.

Hosting a local repository can be a pain though. This means that a server administrator will have to track packages when they get synced to the server and maintain a GPG key on the hosting server as well.

> "Considering the already large size of Debian archive, some people prefer to mirror only parts of it they need. If you want to exclude something, you should exclude architectures.
>
> With the recommended ftpsync, this can be done by editing the ARCH_EXCLUDE variable.

> We strongly advise against excluding the project/, doc/ and other subdirectories. Usually these are minor in size and yet useful to users. Especially project/trace helps very much if there are any mirror issues.
>
> It is possible to use other specially written scripts, but they are usually not necessary, and not recommended for official mirrors.” (Debian, 2016)

The Debian team does provide a mirror script that makes it easy to set up a mirror. This helps with large deployments, locations where there are many client machines and a few servers all running Debian. This is especially useful if there is limited Internet bandwidth available to the network.

Server maintenance is another problem altogether: say someone configured the server initially, yet they did not properly document their steps. How is the new administrator going to have any clue as to how the server was configured? This is a common problem, but all the new administrator needs to do is to look at the installed packages, then look over the config files of the installed packages. Considering that most config files are well commented, it makes the new administrator’s job easy.

References

Anthony, S. (2014, October 3). *IBM unveils new Power8 servers in last-gasp effort to battle Intel's x86 dominion*. Retrieved from Extreme Tech: http://www.extremetech.com/extreme/191453-ibm-unveils-new-power8-servers-in-last-gasp-effort-to-battle-intels-x86-dominion

Debian. (2016, April 24). *Setting up a Debian archive mirror*. Retrieved from Debian: https://www.debian.org/mirror/ftpmirror

Free Software Foundation. (2016, January 01). *What Is Free Software? - GNU Project - Free Software Foundation.* Retrieved from GNU Project - Free Software Foundation.: http://www.gnu.org/philosophy/free-sw.en.html

FSF Webmasters. (2016, April 01). *Microsoft's Software Is Malware - GNU Project - Free Software Foundation*. Retrieved from GNU Project - Free Software Foundation: http://www.gnu.org/proprietary/malware-microsoft.html

GNU.org. (2015, September 21). *Proprietary Software - GNU Project - Free Software Foundation*. Retrieved from The GNU Operating System and the Free Software Movement: https://www.gnu.org/philosophy/proprietary.html

Hachman, M. (2015, October 1). *The price of free: how Apple, Facebook, Microsoft and Google sell you to advertisers*. Retrieved from PCWorld: http://www.pcworld.com/article/2986988/privacy/the-price-of-free-how-apple-facebook-microsoft-and-google-sell-you-to-advertisers.html

IPfire. (2016). *About IPFire The Open Source Firewall Distribution*. Retrieved from About IPFire: http://www.ipfire.org/features

Lee, M. (2015, December 28). *BibMe Free MLA Bibliography & Citation Maker.htm*. Retrieved from The Intercept.

Leonhard, W. (2016, February 3). *New details emerge about forced Windows 10 upgrade -- and how to block it*. Retrieved from InfoWorld: http://www.infoworld.com/article/3029613/microsoft-windows/new-details-emerge-about-forced-windows-10-upgrade-and-how-to-block-it.html

Libreboot. (n.d.). *Answers to Frequently Asked Questions about libreboot*. Retrieved from Libreboot project: https://libreboot.org/faq

Moody, G. (2013, June 17). *How Can Any Company Ever Trust Microsoft Again?* Retrieved from Computer World UK: http://www.computerworlduk.com/blogs/open-enterprise/how-can-any-company-ever-trust-microsoft-again-3569376/

Raptor Engineering. (2016). *TALOS*. Retrieved from Raptor Engineering.

Trisquel. (2009, May 5). *What is Trisquel*. Retrieved from Trisquel GNU/Linux: https://trisquel.info/en/wiki/documentation

Vandewege, W., Garrett, M., & Stallman, R. M. (2014, June 19). *"Active Management Technology": The obscure remote control in some Intel hardware*. Retrieved from Free Software Foundation: https://fsf.org/blogs/community/active-management-technology